MFP implementation for Image Segmentation using EfficientNet and ViT-B

Aniket Pandey (202411001)[1] **and Vishnu Vardhan(202411071)**[1]
[1] Dhirubhai Ambani University, Gandhinagar, Gujarat

1 Introduction

Image segmentation is a fundamental task in computer vision that involves partitioning an image into multiple segments or regions, each corresponding to distinct objects or areas of interest, to simplify and enhance the analysis of visual data. This process is crucial for a wide range of applications, including autonomous driving, medical imaging, object detection, and scene understanding, as it enables machines to interpret and understand images at a pixel level. Traditional segmentation methods relied on low-level features like color, texture, or edges, but they often struggled with complex scenes due to limited contextual understanding. With the advent of deep learning, modern image segmentation techniques, such as semantic segmentation, instance segmentation, and interactive segmentation, have significantly advanced, leveraging convolutional neural networks (CNNs) and transformer-based architectures to achieve higher accuracy and robustness. These methods can automatically learn hierarchical features from raw data, enabling precise delineation of objects even in challenging scenarios with varying lighting, occlusions, or cluttered backgrounds. Interactive segmentation, in particular, has gained attention for its ability to incorporate user inputs, such as clicks or scribbles, to refine segmentation results iteratively, making it highly effective for tasks requiring human-in-the-loop precision.
This project focuses on reproducing the core concepts of the paper "MFP: Making Full Use of Probability Maps for Interactive Image Segmentation" [1], which introduces the MFP framework to enhance click-based interactive image segmentation by leveraging previous probability maps. We implemented the MFPNet architecture using a ViT-B backbone and EfficientNet-B0 backbone, incorporating probability map modulation, late fusion, and a segmentation head. The model was trained on a subset of the COCO-LVIS dataset and tested on the Berkeley dataset, aligning with the paper's evaluation setup. Due to computational constraints, we omitted the click-based interaction and recursive training, focusing on validating the framework's ability to improve segmentation accuracy through modulated probability maps. Our implementation includes a detailed analysis of the architecture via block diagrams and qualitative assessments, aiming to verify the paper's claim of enhanced segmentation performance by fully exploiting probability map information.

2 Scope of Reproducibility

This reproducibility study focuses on implementing and evaluating the core components of the MFP framework proposed in the paper "MFP: Making Full Use of Probability Maps for Interactive Image Segmentation" (CVPR 2024). We reproduce the MFPNet architecture using a ViT-B backbone, incorporating probability map modulation, late fusion, and the segmentation head as described. The implementation is trained on a subset of the COCO-LVIS dataset and tested on the Berkeley dataset, aligning with the paper's evaluation setup. Due to computational constraints, we omit the click-based interactive component (e.g., click map C_t) and recursive training, focusing instead on the

network's ability to leverage previous probability maps for segmentation. We evaluate the model using qualitative comparisons and aim to verify the paper's claim of improved segmentation by exploiting probability maps, while acknowledging deviations in training scale and interaction simulation.

3 Methodology

The methodology for adapting the MFP algorithm from interactive to non-interactive image segmentation, inspired by the CVPR 2024 paper by Lee et al., involves modifying the framework to eliminate user click inputs while leveraging the probability map modulation and network architecture. Initially, the dataset, comprising LVIS for training and Berkeley for testing, is preprocessed with transformations like resizing and normalization to ensure consistency. The ViT-B and EfficientNet-B0 backbones extract features from input images, replacing click maps with an initial zeroed probability map, as no user annotations guide the process. The probability map modulation, originally driven by click-based gamma correction, is adapted to enhance foreground and background regions using a fixed threshold (e.g., 0.7) and gamma values (e.g., 2.0), applied uniformly across the image to refine shape details autonomously.
The network architecture fuses backbone features with modulated probability features using convolutional blocks, followed by a segmentation head to produce a probability map, thresholded to generate the final mask. Recursive training is replaced with standard supervised learning, minimizing a combined BCE and Dice loss over 10 epochs using the Adam optimizer. The model is evaluated on the Berkeley dataset, with performance assessed via mIoU and accuracy metrics, ensuring effective segmentation without interactive inputs by fully exploiting learned probability map representations.

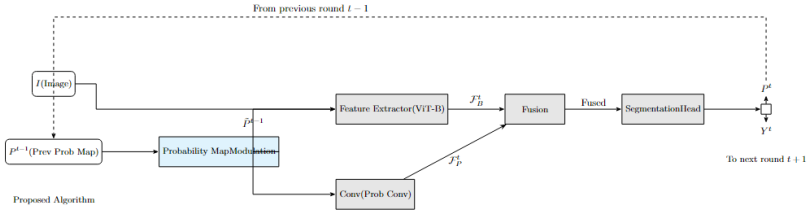

Figure 1. Proposed Algorithm

3.1 Dataset

The dataset for the non-interactive image segmentation adaptation of the MFP algorithm consists of LVIS for training and Berkeley for testing. LVIS, a large-scale dataset, includes approximately 100,000 images with 1.2 million instance masks across diverse object categories, though only 1/20th (around 5,000 images) is used to manage computational constraints, with annotations providing detailed segmentation polygons. The Berkeley dataset, used for evaluation, contains 96 test images with 100 high-quality object masks, offering precise ground-truth segmentations derived from .mat files. Both datasets undergo preprocessing with transformations like resizing to 224x224 pixels and normalization to ensure compatibility with the EfficientNet-B0 backbone, supporting robust training and accurate performance assessment via mIoU and accuracy metrics.

3.2 Computational Requirements

All experiments and model evaluations were conducted using Kaggle's free GPU runtime environment. The key system specifications are as follows:

- **Platform**: Kaggle Notebooks
- **CPU**: Dual-core Intel Xeon (2.3 GHz)
- **RAM**: 13 GB
- **GPU**: NVIDIA Tesla P100 (16 GB VRAM)
- **Disk**: 20 GB available persistent storage (per session)
- **Python Version**: 3.11

4 Results

Table 1. mIoU Results for Different Backbones after 5 and 10 Epochs

Backbone Used	mIoU after 5 Epochs	mIoU after 10 Epochs
ViT-B	0.1086	0.2326
EfficientNet-B0	0.0289	0.1057

When trained on EfficientNet-B0 backbone, we observed the following segmentations results:

Figure 2. EfficientNet-B0 result after 5 epochs

Figure 3. EfficientNet-B0 result after 10 epochs

When trained on ViT-B backbone, we observed the following segmentations results:

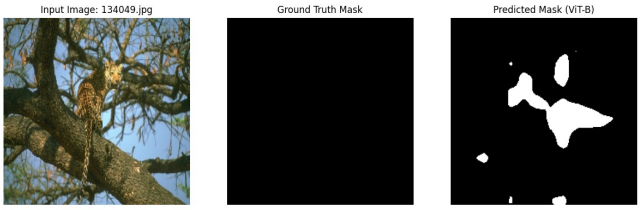

Figure 4. ViT-B result after 5 epochs

Figure 5. ViT-B result after 10 epochs

5 Discussion

The segmentation performance of the ViT-B model improves significantly over training epochs. After 5 epochs, the predicted masks contain incomplete and scattered object regions, indicating underfitting and limited learning. By the 10th epoch, the predicted masks show clearer, more coherent object boundaries that align better with the expected object shapes, suggesting better feature learning and convergence of the model. This demonstrates the benefit of extended training for achieving accurate segmentation. Same can be said for EfficientNet-B0 backbone however its performance is worse than ViT-B backbone. Both the backbones need a bigger training set and more epochs to train efficiently and segment images properly.

5.1 What was Easy

The authors' GitHub repository (https://github.com/cwlee00/MFP) provided clear, step-by-step instructions and well-documented code, facilitating pipeline reimplementation.

5.2 What was Difficult

Trying to implement the "interactive" part in the paper was a challenge that we unfortunately could not overcome due to computational and time constraints. Adding click probability meant we had to include User Interface and we didn't have the resources for that. Hence, we had to resolve to a simpler version of the model that only deals with image segmentation for lesser number of epochs. The training dataset COCO-LVIS is around 26 GB and contains 99K images. Training them all for multiple epochs was too time and memory consuming which our devices couldn't afford hence he took small subsets of training data to train our model on. Since we trained the model on a filtered down dataset and lesser number of epochs, it led to poor test accuracy with suboptimal segmentation.

5.3 Contribution

Each author contributed to specific components of the project. Vishnu Vardhan (202411071) implemented the image segmentation using EfficientNet and Aniket Pandey (202411001) has implemented using ViT-B and obtained results as shown above. Both the authors collaborated on the evaluation, analysis, and writing of this report.

6 Acknowledgement

This work was conducted as part of a course project under the guidance of Dr. Rachit Chhaya at Dhirubhai Ambani University. We thank the authors of the MFP paper for their responsive communication and open-source contributions.

7 Future Work

We will try to improve and implement our model again to come close to the model performance in the paper. Next, we will try to incorporate a novel architecture named "Kolmogorov Arnold Networks" (KANs) to replace the activation function ReLU in our MFPNet model with a learnable activation function and observe its performance. We already tried it before but it didn't get satisfactory results hence we did not include it in our report.

References

1. C. Lee, S.-H. Lee, and C.-S. Kim. "MFP: Making Full Use of Probability Maps for Interactive Image Segmentation." In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. June 2024, pp. 4051–4059.